Can We Apply Projection Based Frequent Pattern Mining Paradigm to Spatial Co-location Mining?

Yan Huang, Liqin Zhang, and Ping Yu

Department of Computer Science and Engineering,
University of North Texas,
P.O. Box 311366, Denton, Texas 76203
{huangyan, lzhang, py0003}@unt.edu

Abstract. A co-location pattern is a set of spatial features whose objects are frequently located in spatial proximity. Spatial co-location patterns resemble frequent patterns in many aspects. Since its introduction, the paradigm of mining frequent patterns has undergone a shift from a generate-and-test based frequent pattern mining to a projection based frequent pattern mining. However for spatial datasets, the lack of a transaction concept, which is critical in frequent pattern definition and its mining algorithms, makes the similar shift of paradigm in spatial co-location mining very difficult. We investigate a projection based co-location mining paradigm. In particular, we propose a projection based co-location mining framework and an algorithm called **FP-CM**, for **FP-growth Based Co-location Miner**. This algorithm only requires a small constant number of database scans. It out-performs the generate-and-test algorithm by an order of magnitude as shown by our preliminary experiment results.

1 Introduction

We focus on a recent spatial data mining problem: finding spatial features that tend to be located in spatial proximity. This problem is also referred to as *spatial co-lcoation patterns mining* [7, 4, 2, 10, 9]. Let $\mathcal{F} = \{f_1, f_2, \ldots, f_l\}$ be a set of spatial features. consider a number of l spatial datasets $\{SD_1, SD_2, \ldots, SD_l\}$, such that $SD_i, i \in [1, l]$ contains all and only the objects that have the spatial feature f_i, Let $\mathcal{R}$ be a given spatial neighbor relation (e.g. distance less than 1.5 miles). A set of spatial features $X \subseteq \mathcal{F}$ is a co-location if its value $im(X)$ of an interesting measure, is above a threshold min_im. The problem of finding the complete set of co-location patterns is called the co-location mining problem. Mining *spatial co-location patterns* is an important spatial data mining task with broad applications.

Spatial co-location patterns resemble frequent patterns [5], a more general problem of mining association rules [1] in many aspects. Since its introduction, the problem of mining frequent patterns from large databases, has been subject

T.B. Ho, D. Cheung, and H. Liu (Eds.): PAKDD 2005, LNAI 3518, pp. 719–725, 2005.

of numerous studies. The paradigm of frequent pattern mining algorithms has undergone a fundamental shift from generate-and-test approaches [1] to projection based approaches [5]. Projection based approaches have major advantages over generate-and-test approaches and avoids multiple database scans by compressing transactional data into compact structures. However, the lack of prematerialized transactions becomes a major obstacle in adopting projection based algorithms in spatial co-location pattern mining. A natural question to ask is: can we push the same paradigm shift for mining spatial co-location patterns?

Many algorithms for co-location mining proposed in literature [7, 4, 10, 9, 3] employ an generate-and-test co-location mining paradigm, which utilizes the anti-monotone property of interestingness measures. In a clustering-based map overlay approach[4, 3], every spatial feature is treated as a map layer and point-data in each layer are clustered into regions. In a reference feature based approach [7], transactions are created according to different algorithms, then a level wise algorithm is applied . Under this model, a frequent pattern based algorithm can be applied straightforwardly due to the fact that the interestingness measure is defined based on the generated transactions. In distance based approaches [9, 10], the number of instances for each spatial feature set is used to define the interestingness measure. In an event centric model [10], a participation index was defined as the interestingness measure. The participation index of a pattern is defined as the minimal participation ratio of the objects of each feature in the pattern.

The contribution of this work is to study how to use a projection based paradigm for event based spatial co-location pattern mining (CM). We proposed a projection based framework for CM, which can incorporate any fast frequent pattern mining algorithm. In particular, we developed an FP-growth based algorithm for spatial co-location mining(FP-CM) based on the proposed framework. We provide preliminary experiment results to show that the FP-CM is an order of magnitude faster than the generate-and-test algorithm *Co-location Miner*.

Paper Outline: Section 2 recalls important concepts of co-location and frequent pattern mining. Section 3 proposes our projection based FP-CM framework and a FP-growth based co-location mining algorithm. We present the preliminary experimental results in section 4 and summarize our work and present future work in section 5.

2 Background

We review basic concepts of co-location patterns, a traditional generate-and-test co-location mining algorithm, and a projection based frequent pattern mining algorithm in this section.

In an event centric model [10], a participation index was defined as the interestingness measure. For a set of spatial features $X \subseteq \mathcal{F}$, a set of objects $\{o_1, o_2, \ldots, o_k\}$ is an *instance* of X iff $(\forall i, i \in [1, k], o_i \in SD_i)$ and $(\forall i \forall j, 0 < i < j \leq k, (o_i, o_j) \in \mathcal{R})$. The *participation ratio* $pr(f, X)$ of a feature f in a pattern X is defined as:

$$pr(f, X) = \frac{\text{number of objects of f that participate in any instance of } X}{\text{total number of objects of } f}$$

The *participation index* of a pattern X is defined as: $pi(X) = \min_{\forall f \in X}\{pr(f, X)\}$. Because of the downward closure property of the participation index [10], a generate-and-test mining paradigm was employed by previous algorithms, e.g. *Co-location Miner*. This approach generates the candidate size $k+1$ co-location set based on the size k co-location set. The candidate size $k+1$ co-location set includes all and only those size $k+1$ spatial feature set whose size k subsets are all co-locations. Then it uses spatial joins on the instances of size k co-locations to generate the instances of the size $k+1$ candidates and calculate the participation indexes for them. It prunes false candidates before starting the next iteration.

Projection based frequent pattern mining utilizes a highly condensed prefix-tree structure to compress frequent patterns and employs a pattern fragment growth method for mining the complete set of frequent patterns from the prefix-tree. Due to the reduced number of database scans, this algorithm is very fast compared with traditional generate-and-test algorithms [5]. (We refer readers to [5] for the details of the FP-growth algorithm). However, a FP-growth based algorithm can not be used directly in spatial co-location mining due to the lack of transactions in spatial datasets. Transactionizing spatial datasets and establishing the relationship between *support* and *participation index* to develop a complete and correct projection based co-location mining algorithm is non-trivial.

3 A Projection Based Co-location Mining Framework

Our proposed framework is shown in Figure 3. A transactional database TD_i is created for each spatial feature f_i. Any fast maximal frequent pattern mining algorithm may be applied to each transactional database TD_i to find maximal frequent patterns $MFP_s = \cup_{i=1...K} MFP_i$ using a support threshold $min\_sup = min\_pi$. The mined maximal frequent patterns MFP_i are combined by a pattern combining component to generate a superset of all the co-location patterns. Finally, a pattern filtering component filters out the false candidate co-locations.

Based on the projection based framework, we develop an algorithm called **FP-CM**, for **FP-growth Based Co-location Miner**. This algorithm consists of four components: transactionization, maximal frequent pattern mining, combining patterns, and pattern filtering.

1. *Transactionization (step 2)*
   For each spatial feature f, we create a transactional database TD_f as follows. For each object o of f, a transaction containing all other spatial features whose object(s) is(are) within neighbor $\mathcal{R}$ of o is created.
2. *Maximal Frequent Pattern Mining (step 3)*
   For each transactional database TD_f, we find all maximal frequent patterns

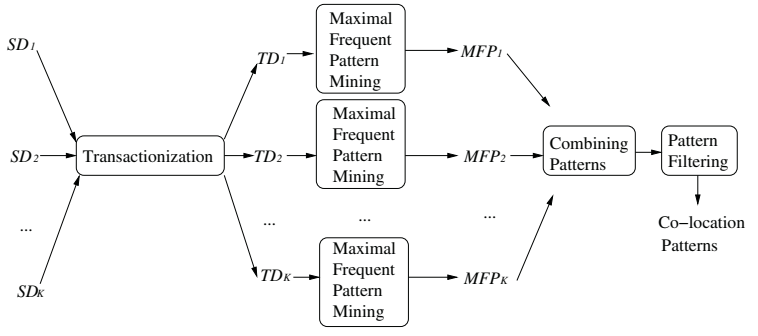

Fig. 1. Projection Based Co-location Pattern Mining Framework

Algorithm 1. FP-CM

1: **for** $i = 1$ to K **do**
2: $TD[i] \leftarrow transactionize(SD[1], SD[2], \ldots, SD[K])$;
3: $MFP[i] \leftarrow FP - growth(TD[i], min\_pi)$;
4: **end for**
5: $i \leftarrow 1$;
6: $C[1] \leftarrow \{1, 2, \ldots, K\}$;
7: **while** $C[i] \neq \emptyset$ **do**
8: $C[i+1] \leftarrow apriori\_gen(C[i])$ /*refer to [1]*/;
9: $C[i+1] \leftarrow prune(C[i+1], MFP[1], MFP[2], \ldots, MFP[K])$;
10: $i \leftarrow i + 1$;
11: $C \leftarrow C \cup C[i]$;
12: **end while**
13: $P \leftarrow multi - way - spatial - join - prune(C, min\_pi)$;
14: **return** P;

based on the FP-growth frequent pattern mining algorithm using a support threshold $min\_sup_f = min\_pi$ in this step.

3. *Combining Patterns (step 5-12)*
The basic structure of the combining pattern step is the level-wise structure of CM [10]. However, it does not require expensive spatial joins to calculate participation indexes. Instead, it consults the MFPs to prune the majority of the false candidate patterns. This step will produce a superset of the true co-location patterns to feed to the next pattern filtering step.
The prune step (step 9) works as follows. For each candidate pattern C, $\forall f \in C$, if MFP_f does not contain a superset of $(C - f)$, then C is pruned. This will not falsely delete any true patterns since $pr(f, X) \geq min\_pi$ implies $(C - f)$ is frequent and should have a superset in MFP_f.
4. *Pattern Filtering (step 13)*
Once we reduce the total number of candidate co-location patterns from

$2^{\#features}$ to a small superset of the true co-location patterns, we can use hash-based spatial join techniques [6] and multi-way spatial joins [8] to filter the patterns. We hash spatial datasets into buckets using a grid [6] and then use a multi-way spatial join which is based on a backtracking search heuristic [8] to find all the maximal cliques. We keep the list of all the candidate co-location patterns from the previous step and register the cliques to their corresponding candidate co-location patterns. Finally we calculate the actual participation indexes for each candidate co-location pattern and return the set of all co-location patterns found.

The FP-CM algorithm requires a small number of database scans. One database scan is required to transactionize the spatial data, then FP-growth based maximal pattern mining requires two or a few database scans depending on the average size of the FP-trees. Combining patterns involves only spatial features and the maximal frequent pattern sets and usually is a memory based step. Finally, the pattern filtering step using gridding and multi-way spatial joins requires two more database scans. So the total number of database scans of FP-CM algorithm is bounded by a small constant.

4 Experiment Results

We implemented both the co-location miner (CM) and FP-growth based co-location miner (FP-CM) using C++ and all the experiments are carried out on a Pentium IV 2.4GHz machine with 1GB memory, running the Debian linux operating system. Our experiments are extensive and the results are consistent. Limited by space, we only report representative results for various parameters. Our dataset generator is similar to [1].

We use a notation like $|P|50.PS5.|F|100.|I|24k.min\_pi0.4$ to denote an experiment with 50 pre-generated patterns whose average size is 5 and the number of features participating in a pattern is 100, 24k spatial objects, and minimum participation index threshold is 0.4. Since the time for computing size 2 co-locations are the same (one database scan) for both algorithms, we only report the time for calculating size 3 or more co-location patterns.

1. *Effect of thresholds:*
   As Figure 2 (a) shows, FP-CM is much faster than CM for all the threshold range in [0.5,0.2]. The advantage of FP-CM over CM increases when the participation threshold decreases due to the increased number of candidate patterns and associated spatial joins CM has to perform. FP-CM is an order of magnitude faster than CM when the participation index threshold is low.
2. *Effect of total number of Objects:*
   We compare the scalability of the two algorithms when the total number of objects increase from 5k to 50k. As shown in Figure 2 (b), FP-CM is 5 to 40 times faster than CM and the running time of the FP-CM algorithms remains almost the same while the running time of the CM increases dramatically.

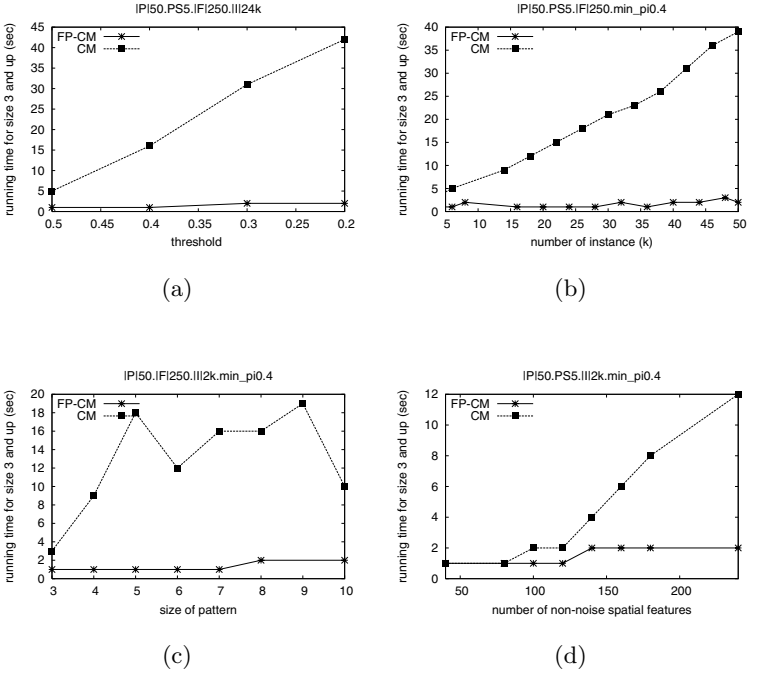

Fig. 2. Performance Comparison of **CM** and **FP-CM**

3. *Effect of Average Maximal Pattern Size:*
   Figure 2 (c) shows the result when the size of the pattern ranges from 3 to 10. FP-CM is 3 to 18 times faster than CM. The running time of FP-CM is stable as the size of the patterns increases while the the running time of CM highly correlates with the total number of co-locations found.
4. *Effect of Number of Patterns:*
   We range the number of non-noise spatial features from 50 to 250 as shown in Figure 2 (d). FP-CM is up-to 12 times faster than CM when the number of non-noise spatial features increases.

5 Conclusion and Future Work

In this paper we proposed a projection based framework for mining spatial co-locations, which is flexible in incorporating any fast maximal frequent pattern mining algorithm developed in literature to help spatial co-location mining. In particular, we developed a complete and correct FP-tree based algorithm for spatial co-location mining. It combines the salient features of FP-tree based maximal frequent pattern mining [5] and fast multi-way spatial joins [8] to reduce the total number of database scans into a small constant. Our experiment results

showed that the FP-CM is an order of magnitude faster than a generate-and-test algorithm *Co-location Miner*.

The proposed projection based co-location mining framework could be treated as a new data-driver partitioning of spatial datasets according to the objects of each spatial features. Compared with traditional spatial partition approaches [11], this approach does not have the problem of combinatorial explosion of temporary candidate patterns needed to be maintained by the algorithm before all the partitions are processed as acknowledged by the authors in [11]. In future work, comparing various partition based co-location mining algorithms would be an interesting and imperative research direction.

References

1. R. Agarwal and R. Srikant. Fast Algorithms for Mining Association Rules. In *Proc. of the 20th Int'l Conference on Very Large Data Bases*, 1994.
2. N.A.C. Cressie. *Statistics for Spatial Data*. Wiley and Sons, 1991.
3. V. Estivill-Castro and I. Lee. Data Mining Techniques for Autonomous Exploration of Large Volumes of Geo-referenced Crime Data. In *Proc. of the 6th International Conference on Geocomputation*, 2001.
4. V. Estivill-Castro and A. Murray. Discovering Associations in Spatial Data - An Efficient Medoid Based Approach. In *Proc. of the Second Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 1998.
5. Jiawei Han, Jian Pei, and Yiwen Yin. Mining frequent patterns without candidate generation. In *ACM SIGMOD Intl. Conference on Management of Data*, 2000.
6. D. J. DeWitt J. M. Patel. Partition Based Spatial-Merge Join. In *Proc. of the ACM SIGMOD Conference on Management of Data*, June 1996.
7. K. Koperski and J. Han. Discovery of Spatial Association Rules in Geographic Information Databases. In *Proc. of the 4th International Symposium on Spatial Databases*, 1995.
8. Nikos Mamoulis and Dimitris Papadias. Multiway spatial joins. *ACM Trans. Database Syst.*, 26(4):424–475, 2001.
9. Y. Morimoto. Mining Frequent Neighboring Class Sets in Spatial Databases. In *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.
10. S. Shekhar and Y. Huang. Co-location Rules Mining: A Summary of Results. In *Proc. 7th Intl. Symposium on Spatio-temporal Databases*, 2001.
11. Xin Zhang, Nikos Mamoulis, David W. Cheung, and Yutao Shou. Fast Mining of Spatial Collocations. In *Proc. of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.